Hardware Simulator Design for MIMO Propagation Channel on Shipboard at 2.2 GHz

Bachir Habib, Hanna Farhat, Gheorghe Zaharia, Ghaïs El Zein

Hardware Simulator Design for MIMO Propagation Channel on Shipboard at 2.2 GHz

Bachir Habib, Hanna Farhat, Gheorghe Zaharia, Ghaïs El Zein

Institute of Electronics and Telecommunications of Rennes, IETR, UMR CNRS 6164, Rennes, 35708, France

Tel: +33 (0)2 23 23 86 04

+33 (0)6 17 57 13 87

Bachir.habib@insa-rennes.fr

www.ietr.fr

A wireless communication system can be tested either in actual conditions or with a hardware simulator reproducing actual conditions. With a hardware simulator it is possible to freely simulate a desired radio channel and making it possible to test "on table" mobile radio equipments. This paper presents new architectures for the digital block of a hardware simulator of MIMO propagation channels. This simulator can be used for LTE and WLAN IEEE 802.11ac applications, in indoor and outdoor environments. However, in this paper, specific architectures of the digital block of the simulator for shipboard environment are presented. A hardware simulator must reproduce the behavior of the radio propagation channel. Thus, a measurements campaign has been conducted to obtain the impulse responses of the shipboard channel using a channel sounder designed and realized at IETR. After the presentation of the channel sounder, the channel impulse responses are described. Then, the new architectures of the digital block of the hardware simulator, implemented on a Xilinx Virtex-IV FPGA are presented. Moreover, the measured impulse responses are implemented in the simulator. The accuracy, the occupation on the FPGA and the latency of the architectures are analyzed.

Channel sounder; hardware simulator; radio channel; MIMO; FPGA

1. Introduction

Wireless communication systems may offer high data bit rates by achieving a high spectral efficiency using Multiple-Input Multiple-Output (MIMO) techniques. MIMO systems make use of antenna arrays simultaneously at both transmitter and receiver sites to improve the capacity and/or the system performance. However, the transmitted electromagnetic waves interact with the propagation environment. Thus, it is necessary to take into account the main propagation parameters during the design of the future communication systems. The current communication standards indicate a clear trend in industry toward supporting MIMO functionality. In fact, several studies published recently present systems that reach a MIMO order of 8×8 and higher [1]. This is made possible by advances at all levels of the

communication platform, as the monolithic integration of antennas [2] and the simulator platforms design [3].

The objectives of our work mainly concern the channel models and the digital block of the simulator. The design of the RF blocks was completed in a previous project [4].

The channel models can be obtained from standard channel models, as the TGn IEEE 802.11n [5] and the LTE channel models [6], or from real measurements conducted with the MIMO channel sounder designed and realized at IETR [7]. Due to the MIMO channel sounder, the hardware simulator will be able to use real measurements. The channel sounder has a 100 MHz bandwidth and 200 MHz sampling frequency at a carrier frequency of 2.2 GHz or 3.5 GHz.

Recently, the channel sounder was used during a measurement campaign on the ferryboat (Armorique of Brittany Ferries) shown in Figure 1.

Figure 1. Ferryboat Armorique of Brittany Ferries

In the MIMO context, little experimental results have been obtained regarding time-variations, partly due to limitations in channel sounding equipment [8]. However, theoretical models of impulse responses of time-varying channels can be obtained using Rayleigh fading [9, 10].

Tests of a radio communication system, conducted under actual conditions are difficult, because tests taking place outdoors, for instance, are affected by random movements or even by the weather. In addition, a test conducted in one environment (city A) does not fully apply to another environment (city B). Usually, under actual conditions, it is difficult to achieve the most difficult radio propagation conditions in order to determine the performance of a wireless communication system.

However, with hardware simulators, it is possible to very freely simulate desired types of radio channels. Moreover, a hardware simulator provides the necessary processing speed and real time performance, as well as the possibility to repeat the tests for any MIMO system. Thus, a hardware simulator can be used to compare the performance of various radio communication systems in the same desired test conditions.

These simulators are standalone units that provide the fading signal in the form of analog or digital samples. Some MIMO hardware simulators are proposed by industrial companies like Spirent (VR5) [11], Azimuth (ACE), Elektrobit (Propsim F8) [12], but they are quite expensive.

With continuing increase of the Field Programmable Gate Array (FPGA) capacity, entire baseband systems can be mapped onto faster FPGAs for more efficient prototyping, testing and verification. Larger and faster FPGAs permit the integration of a channel simulator along with the receiver noise simulator and the signal processing blocks for rapid and cost-effective prototyping and design verification. As shown in [13], the FPGAs provide the greatest design flexibility and the visibility of resource utilization. They are ideal for rapid prototyping and research use such as testbed [14].

The MIMO hardware simulator realized at IETR is reconfigurable with sample frequencies not exceeding 200 MHz, which is the maximum value for FPGA Virtex-IV. The MIMO channel sounder realized at IETR provides a sample frequency of 200 MHz. Thus, it is compatible with the FPGA Virtex-IV. However, in order to exceed 200 MHz for the sample frequency, more performing FPGA as Virtex-VII can be used [3]. Moreover, [15] presented a study relating the sampling frequency to the occupation on the FPGA. This study shows that doubling the sampling frequency allows us to simulate two channels instead of one in the same time period.

The simulator is able to accept input signals with wide power range, between -50 and 33 dBm, which implies a power control for the input signals. Also, a simple, fast and accurate amplitude estimator method for single sinusoid signals presented in [16, 17] can be used to estimate the FPGA signal parameters and compare it with real measured signal.

At IETR, several architectures of the digital block of a hardware simulator have been studied, in both time and frequency domains [4, 18]. Other researchers [19] presents a new method based on determining the parameters of a channel simulator by fitting the space time-frequency cross-correlation matrix of the simulation model to the estimated matrix of a real-world channel. This solution shows that the obtained error can be important.

Typically, wireless channels are commonly simulated using finite impulse response (FIR) filters, as in [4, 20, 21]. The FIR filter form a convolution between a channel impulse response and a fed signal in such a manner that the signal delayed by different delays is weighted by the channel coefficients, i.e. tap coefficients, and the weighted signal components are summed up. The channel coefficients are periodically modified to reflect the behavior of an actual channel. Nowadays, different approaches have been widely used in filtering, such as distributed arithmetic (DA) and canonical signed digits (CSDs) [22].

However, using a FIR filter in a channel simulator has a limitation. With a FPGA Virtex-IV, it is impossible to implement a FIR filter operating at a sample frequency of 200 MHz with more than 192 multipliers (impulse response with more than 192 taps).

To simulate an impulse response with more than 192 taps, the Fast Fourier Transform (FFT) module can be used. With a FPGA Vitrex-IV, the size N of the FFT module can reach 65536 samples. Thus, several frequency domain architectures have been considered and tested [4, 19]. Moreover, a proposed VLSI implementation shows that for high order MIMO arrays, frequency domain architectures are highly modular and scalable by design.

In this paper, we present a study of two alternative approaches. The first approach performs in frequency domain, while the second approach operates in time domain and is based on FIR filter.

The main contributions of the paper are:

- The previous considered frequency domain architectures operate correctly only for signals with a number of samples not exceeding the size of the used FFT block. Thus, in this study, a new frequency architecture [23] avoiding this limitation and a new time domain architecture are both tested for a shipboard environment.
- The time domain architecture presented in [18, 20] determines an occupation of 11 % to 13 % of slices on the FPGA for one SISO channel. However, in this paper, we present a time domain architecture with an occupation of 5 % for one SISO channel and up to 80 % for a MIMO 4×4 systems.
- In general, the channel impulse responses can be presented in baseband with its complex envelope, or as a real signal with limited band between $f_c - B/2$ and $f_c + B/2$, where f_c is the carrier frequency and B is the bandwidth. In this paper, to eliminate the complex multiplication and the f_c, the hardware simulation operates between Δ and $B + \Delta$, where Δ depends on the band-pass filters (RF and IF). The value Δ is introduced to prevent the overlap of the positive and the negative sides of the frequency response. In

addition, the use of a real impulse response allows the reduction by 50% of the size of the FIR filters. Thus, within the same FPGA, more SISO channels (hence, larger MIMO channels) can be simulated.

- Tests have been made for indoor [24], outdoor [25] and vehicular [26] environments using standard channel models. However, in this paper, tests are made for a shipboard environment with real channel measurements realized with the channel sounder for 2×2 MIMO channels. Moreover, time-varying channels are obtained using Rayleigh fading.
- In this study, several improvement solutions are presented; studies are made relating the number of bits used for the samples of the channel impulse response to the relative error at the outputs in order to identify the best trade-off between the occupation on the FPGA and the accuracy.

The rest of this paper is organized as follows. Section 2 presents the channel models used to test the proposed architectures. Section 3 describes the new architectures of the simulator in frequency and time domain respectively. The prototyping platform used to implement these architectures and the occupation on the FPGA for the implementation of each architecture are also described. Section 4 presents the accuracy of the output signals when measured impulse responses are used in the digital block of the hardware simulator. Section 5 presents some improvement solutions to reduce the error, the latency and the occupation on the FPGA. The accuracy of the new architecture is also analyzed. Lastly, Section 6 gives concluding remarks and prospects.

2. Channel Models

A MIMO propagation channel is composed of several time-variant correlated SISO channels. Figure 2 illustrates a MIMO channel with $N_T = 2$ transmit antennas and $N_R = 2$ receive antennas.

For this MIMO channel, the received signal $y_j(t,\tau)$ can be calculated using a convolution in time domain:

$$y_j(t,\tau) = x_1(\tau) * h_{1j}(t,\tau) + x_2(\tau) * h_{2j}(t,\tau) \ , j = 1,2 \quad (1)$$

The associated spectrum is calculated by the Fourier transform (using FFT modules):

$$Y_j(t,f) = X_1(f).H_{1j}(t,f) + X_2(f).H_{2j}(t,f), \ j = 1,2 \quad (2)$$

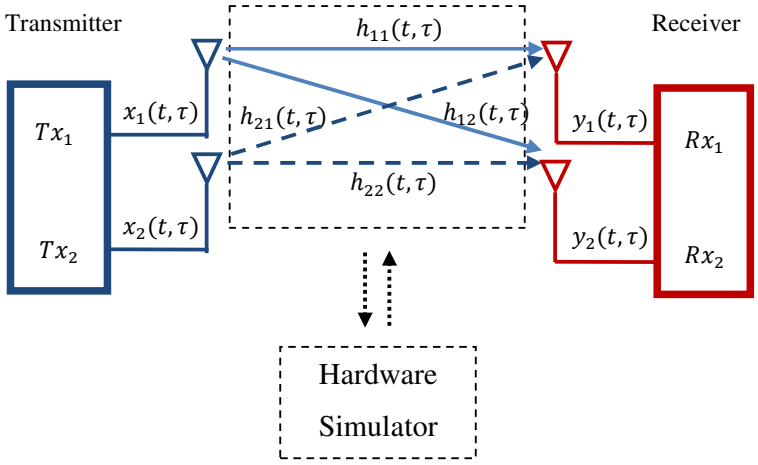

Figure 2. MIMO channel (2×2 SISO channels)

According to the considered propagation environments, Table 1 summarizes some useful parameters for LTE standard, WLAN 802.11ac standard and channel sounder signals for a specific environment on the Armorique which is presented in Figure 1.

Table 1. Simulator Parameters

	Type	Cell size	$W_{t\,eff}$ (μs)	N_F	W_{tF} (μs)	N_T	W_{tT} (μs)
LTE (B=20 MHz)	Rural	2-20 km	20	2048	40.96	1000	20
	Urban	0.4-2 km	3.7	512	10.24	185	3.7
	Indoor	20-400 m	0.7	256	2.56	35	0.7
802.11ac (B=80 MHz)	Office	40 m	0.35	128	0.64	70	0.35
	Indoor	50-150 m	0.71	512	2.56	142	0.71
	Outdoor	50-150 m	1.16	1024	5.12	232	1.16
Channel sounder (B=100 MHz)	Ship -board	9 m	20.48	512	2.56	200	1

$W_{t\,eff}$ represents the time window of the MIMO impulse responses. The number of samples computed for the frequency domain as:

$$N_F = W_{tF}.f_s \tag{3}$$

and for the time domain as:

$$N_T = W_{tT}.f_s \quad (4)$$

where W_{tF} is the closest value for $W_{t\ eff}$ which is imposed by the size $N_F = 2^n$ of the FFT modules.

Two channel models are considered to cover many types of environments: the TGn channel models (indoor environments), the LTE channel models (outdoor environments). Moreover, using the channel sounder realized at IETR, measured impulse responses are obtained for specific environments.

In this study, measured complex impulse responses of the MIMO propagation channel obtained in a shipboard indoor metallic environment were used to supply the digital block of the channel simulator.

2.1. TGn Channel Model

TGn channel models [5] represent a set of 6 profiles, labeled A to F, which cover all the scenarios for WLAN applications. Each model has a number of clusters. Each cluster corresponds to specific tap delays, which overlaps each other in certain cases. 802.11ac signals are considered with a bandwidth of 80 MHz. The sampling frequency and period are f_s = 180 MHz and $T_s = 1/f_s$ respectively. The relative power of each tap of the impulse response for all TGn channel models are presented in [5] by taking the LOS (Line-Of-Sight) path as reference.

2.2. LTE Channel Model

LTE channel models are used for mobile wireless applications. A set of 3 channel models is used to simulate the multipath fading propagation conditions. A detailed description is presented in [6].

2.3. Measurement Data

Impulse responses of a MIMO channel can be obtained from measurements by using a time domain channel sounder designed and realized at the IETR [7]. Several measurement campaigns were carried out for indoor and outdoor environments. Recently, a measurements campaign was carried out in order to obtain measured MIMO impulse responses for a

shipboard environment. These MIMO impulse responses will be used by the hardware simulator.

The channel sounder uses a periodic pseudo random binary sequence. It has 11.9 ns temporal resolution for 100 MHz sounding bandwidth. The carrier frequencies are 2.2 GHz and 3.5 GHz. The synchronization between the transmitter and the receiver is achieved with highly stable 10 MHz rubidium oscillators. For the shipboard measurement campaign at 2.2 GHz, a Uniform Linear antenna Array (ULA) and a Uniform Rectangular antenna Array (URA) were used for the transmitter (T_x) and the receiver (R_x) respectively, to characterize the double directional channel on a 120^o beam-width in the horizontal plan.

Figure 3 presents the propagation channel environment and the position of T_x and R_x.

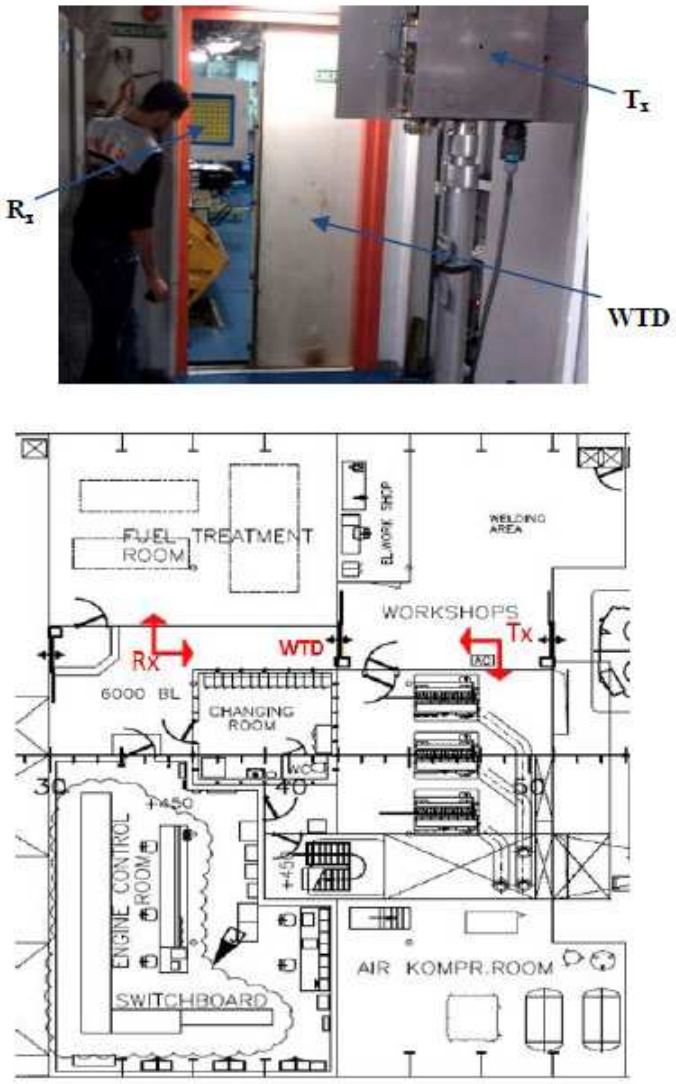

Figure 3. The propagation channel inside the shipboard

This environment is located at floor 2 of the shipboard presented in Figure 1. The WTD is a watertight door. The measurements are made with closed WTD.

To use less multipliers for the digital block of the simulator, the complex impulse responses will be converted to real impulse responses. The notion of complex envelope is used to define, for a real band-limited signal, a complex signal equivalent in baseband. An impulse response can then be expressed by:

$$h(t) = a(t).\cos\big(2\pi f_0 t + \varphi(t)\big)$$
$$= a(t).\cos\big(\varphi(t)\big).\cos(2\pi f_0 t) - a(t).\sin\big(\varphi(t)\big).\sin(2\pi f_0 t) \quad (5)$$

where

$$h_p(t) = a(t).\cos\big(\varphi(t)\big) \quad (6)$$

$$h_q(t) = a(t).\sin\big(\varphi(t)\big) \quad (7)$$

Therefore:

$$h(t) = h_p(t).\cos(2\pi f_0 t) - h_q(t).\sin(2\pi f_0 t) \quad (8)$$

where $h_p(t)$ and $h_q(t)$ are the real and imaginary parts of the complex response.

The channel sounder provides the complex envelope $h_{ce}(t)$ of the baseband signal with a B=100 MHz bandwidth and with a center frequency:

$$f_0 = \Delta + \frac{B}{2} \quad (9)$$

The real impulse responses are obtained by:

$$h(t) = h_p(t).\cos(\pi(2\Delta + B)t) - h_q(t).\sin(\pi(2\Delta + B)t) \quad (10)$$

Therefore, we can work with a real impulse response that occupies the band $[\Delta, \Delta + B]$ (Figure 4).

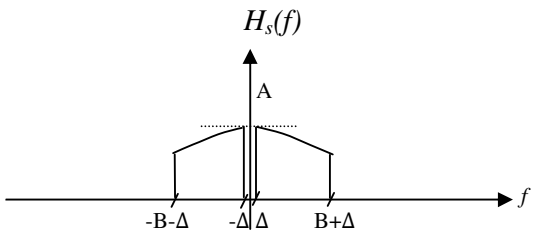

Figure 4. Frequency response used by the hardware simulator

The first results (without normalization) are obtained for a 2×2 MIMO channel. Figure 5 presents the impulse responses given by the channel sounder on $2048T_s$.

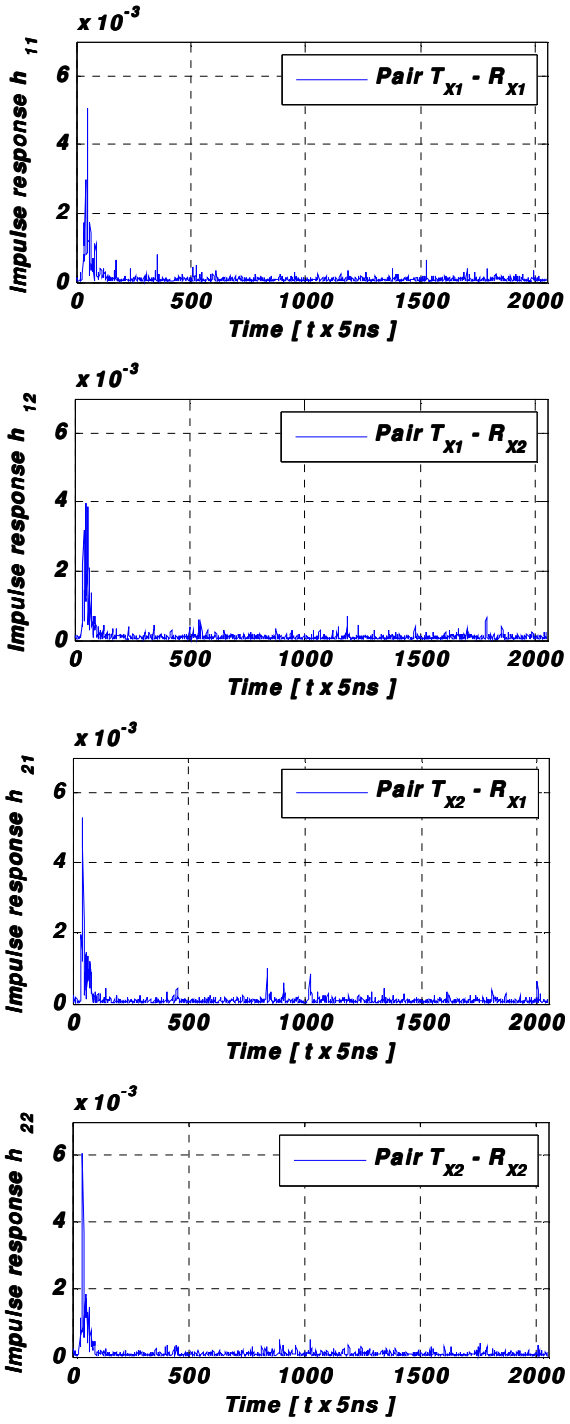

Figure 5. Real impulse responses

Using a FPGA Virtex-IV, the number of multipliers used by a FIR filter is limited to 192. Thus, a high resolution method is proposed [27, 28] in order to estimate the propagation

parameters of this channel and to obtain significant impulse responses (*hDis*) with a limited number of taps and hence a limited number of multipliers for the FIR filter. However, these methods are heavy computation load. Therefore, a new method is proposed which consist of detecting the taps considered as points of change for the sign of the slope of the curve. Figure 6 presents the impulse responses used by the simulator after discrimination, normalization and limitation between 0 and -20 dB of the real impulse responses.

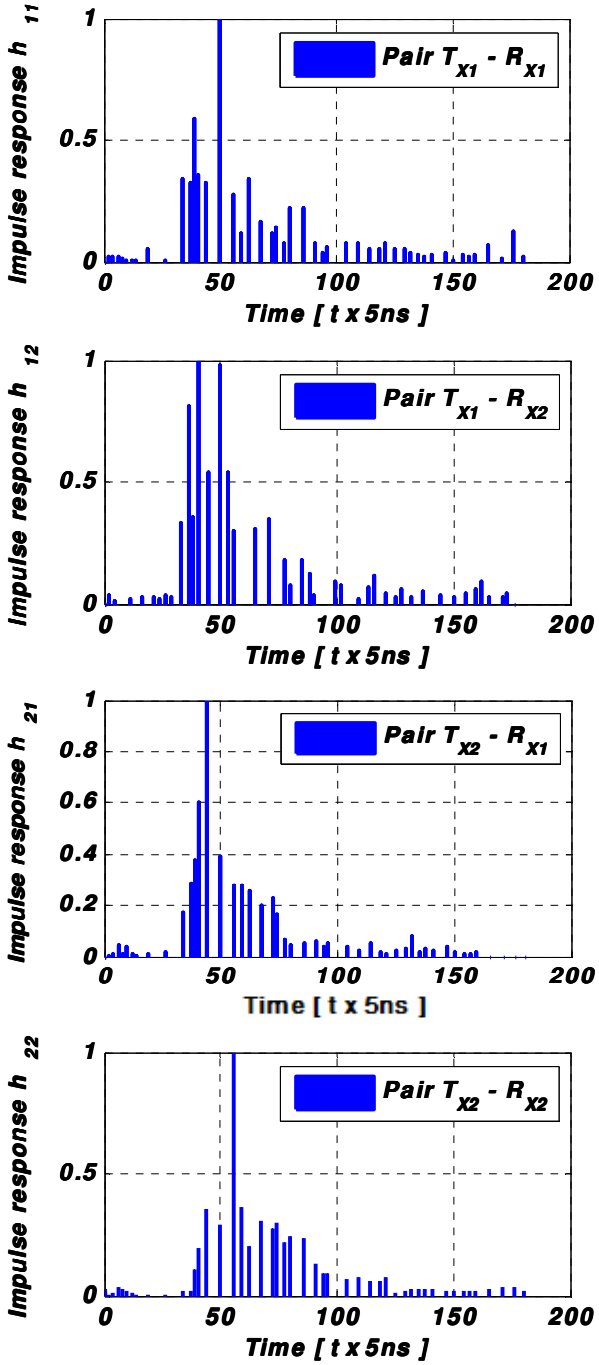

Figure 6. Impulse responses used for the test

2.4. Time-Varying Channels

During the measurements, the channel was time invariant, without people moving in the environment. Therefore, in order to simulate a time-varying channel, a Rayleigh fading method was used.

We define a 2×2 MIMO Rayleigh fading channel [29, 30] using the static impulse responses presented in Figure 6. The MIMO channel matrix H can be characterized by two parameters:

1) The power P_c of constant channel components which corresponds to the Line-Of-Sight (LOS).
2) The power P_s of the channel scattering components which corresponds to the Non-Line-Of-Sight (NLOS).

The ratio P_c/P_s is called Ricean K-factor and it is often represented in decibels.

Assuming that all channel coefficients of the channel matrix H are Rice distributed, the MIMO channel matrix H for each tap can be expressed by:

$$H = \sqrt{P_c}\,.\,H_F + \sqrt{P_s}\,.\,H_v \tag{11}$$

where H_F and H_V are the constant and the scattered channel matrices respectively.

The total received power $P = P_c + P_s$. Therefore:

$$P_c = P\,.\frac{K}{K+1} \tag{12}$$

$$P_s = P\,.\frac{1}{K+1} \tag{13}$$

where K is the Ricean factor and P is the power of each tap given in Figure 6.

Moreover, if we combine (12) and (13) in (11) we obtain:

$$H = \sqrt{P}.\left(\sqrt{\frac{K}{K+1}}H_F + \sqrt{\frac{1}{K+1}}H_V\right) \tag{14}$$

K is equal to zero to obtain a Rayleigh fading channel because the measurements were taken in NLOS, so H can be written as:

$$H = \sqrt{P}.H_V \tag{15}$$

For 2 transmit and 2 receive antennas:

$$H = \sqrt{P}.\begin{bmatrix} X_{11} & X_{12} \\ X_{21} & X_{22} \end{bmatrix} \quad (16)$$

where X_{ij} (i-th receiving and j-th transmitting antenna) are correlated zero-mean, unit variance, complex Gaussian random variables as coefficients of the variable NLOS (Rayleigh) matrix H_V.

To correlate the X_{ij} elements of the matrix X, a product-based model is used. This model assumes that the correlation coefficients are independently derived at each end of the link. It can be expressed by:

$$X = (R_r)^{1/2}.H_{iid}.\left((R_t)^{1/2}\right)^T \quad (17)$$

H_{iid} is a matrix of independent zero means, unit variance, complex Gaussian random variables.

The method for generating the Rayleigh random is:

1) Generation of two sequences (x_{1p} and x_{2p}) of complex Gaussian random variables from 0 to f_d where f_d is the Doppler spread.
2) We take the complex conjugate (x_{1c} and x_{2c}) of these sequences to generate the complex Gaussian random variables for the negative part from $-f_d$ *to 0*.
3) Therefore we obtain $x_1 = x_{1p} + x_{1c}$ and $x_2 = x_{2p} + x_{2c}$.
4) We multiply the above complex Gaussian sequences (x_1 and x_2) with the root of the Doppler Spectrum S for indoor measurements [5] generated from $-f_d$ to f_d :

$$S(f) = \frac{1}{1+9.\left(\frac{f}{f_d}\right)^2} \quad (18)$$

5) To obtain the signals in time domain, we take the IFFT of the two signals above resulting in time domain signals x and y.
6) We define r_i equals to $\sqrt{x^2 + y^2}$.

r_i is an element of the H_{iid} matrix and it is the desired Rayleigh distributed with the required temporal correlation.

R_r and R_t are the receive and transmit correlation matrices, respectively.

We consider α_1 , α_2 the correlations between channels at two receive antennas, but originating from the same transmit antenna (SIMO). β_1 and β_2 are the correlations between

channels at two transmit antennas, but originating from the same receive antenna (MISO). s_1 and s_2 are the cross-correlation between antennas of the same side of the link.

The use of this model has two conditions:

1) $\alpha_1 = \alpha_2 = \alpha$ and $\beta_1 = \beta_2 = \beta$, the correlations between channels at two receive (resp. transmit) antennas are independent from the considered Rx (resp. Tx) antenna, as shown in Figure 7.

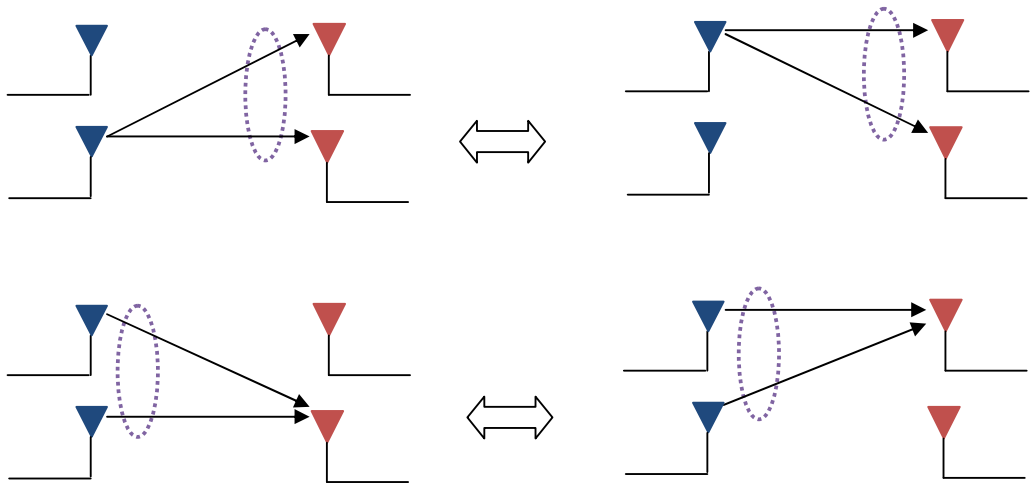

Figure 7. Correlations between channels

2) $s_1 = \alpha \times \beta$ and $s_2 = \alpha^* \times \beta$.

R_t and R_r can be written by:

$$R_t = \begin{bmatrix} 1 & \alpha \\ \alpha^* & 1 \end{bmatrix} \tag{19}$$

$$R_r = \begin{bmatrix} 1 & \beta \\ \beta^* & 1 \end{bmatrix} \tag{20}$$

For the uniform linear array, the complex correlation coefficients α and β are expressed by ρ:

$$\rho = R_{xx}(D) + j.R_{xy}(D) \tag{21}$$

where $D = 2\pi d/\lambda$, $d = 0.5\lambda$ is the distance between two successive antennas, λ is the wavelength and R_{xx} and R_{xy} are the cross-correlation functions between the real parts (equal to

the cross-correlation function between the imaginary parts) and between the real part and imaginary part respectively of the considered correlated angles:

$$R_{xx}(D) = \int_{-\pi}^{\pi} \cos\left(D.\sin\left(\varphi\right)\right).PAS(\varphi).d\varphi \tag{22}$$

$$R_{xy}(D) = \int_{-\pi}^{\pi} \sin\left(D.\sin\left(\varphi\right)\right).PAS(\varphi).d\varphi \tag{23}$$

The calculation of the complex correlation coefficients for each tap is based on the PAS (Power Angular Spectrum) with AS (Angular Spread) being the second moment of PAS. The PAS is found to closely match the Laplacian distribution [31, 32]:

$$PAS(\theta) = \frac{1}{\sqrt{2}\sigma} e^{-|\sqrt{2}\theta/\sigma|} \tag{24}$$

where σ is the standard deviation of the PAS (which corresponds to the numerical value of AS).

3. Digital Block Design of the Hardware Simulator

In this section, improved frequency and time domains architectures are presented and implemented on a FPGA Virtex-IV.

3.1. New Frequency Domain Architecture

The new frequency architecture presented in Figure 8 has been verified with a Gaussian impulse response [23]. It operates correctly for signals with a number of samples exceeding N_F, where $N_F = 2^n$ is the size of the FFT module.

For the shipboard channel models, the largest excess delay is $180T_s$ (Figure 6). Thus, N_F = 256. However, it is mandatory to extend each partial input of N_F samples with a "tail" of NF null samples, as in [23], to avoid a wrong result. Therefore, the FFT/IFFT modules operate with 512 samples.

H is the FFT of h (given in Figure 6). It can be calculated by:

$$H = T_s.h_q.W_q \tag{25}$$

where h_q is h quantified on 32 bits (16 bits for the real part and 16 bits for the imaginary part) and W_q is the quantified version of the matrix used by FFT. Each element of this matrix is quantified on 12 bits.

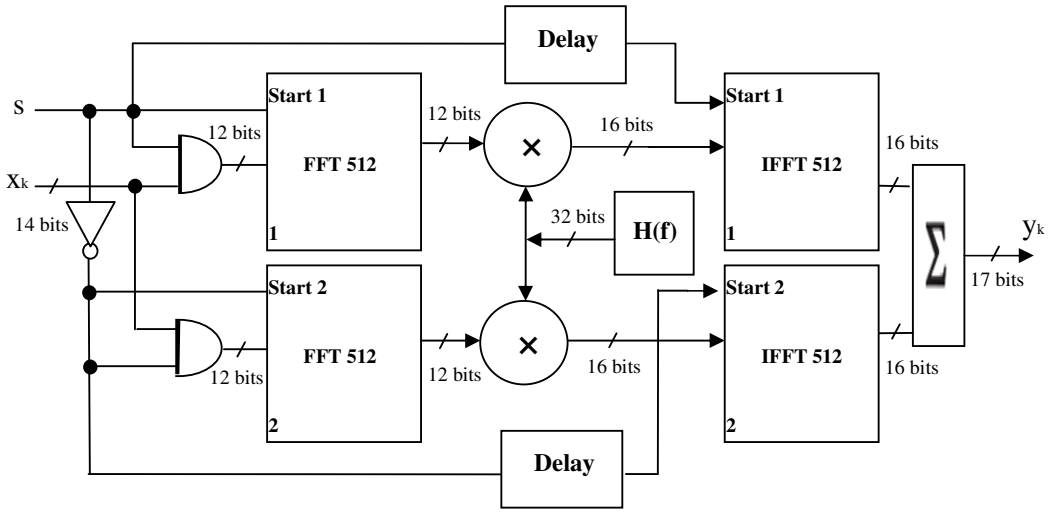

Figure 8. Frequency architecture for a SISO channel

The truncation block is located at the output of the digital adder. It is necessary to reduce the number of bits after the sum of the signals computed by the IFFT blocks to 14 bits. Thus, these samples can be accepted by the digital-to-analog converter (DAC), while maintaining the highest accuracy.

The immediate solution is to keep the first 14 bits. It is a "brutal" truncation.

However, for low voltages of the output of the digital adder, the brutal truncation generates zeros to the input of the DAC. Therefore, a better solution is the sliding window truncation [23] presented in Figure 9 which uses the 14 most effective significant bits.

Figure 9. Sliding window truncation, from 17 to 14 bits

3.2. New Time Domain Architecture

4 SISO channels are implemented. For each SISO channel, the FIR width and the number of used multipliers are determined by the number of taps of each impulse response. For example

for h_{11}, the largest excess delay corresponds to the 178th sample. Thus, N_T = 179 samples. Because h_{11} has 47 paths (non null taps), the FIR filter must use 47 multipliers.

Figure 10 presents a FIR 179 with 47 multipliers. We have developed our own FIR filter [25] instead of using Xilinx MAC FIR filter to make it possible to reload the FIR filter coefficients.

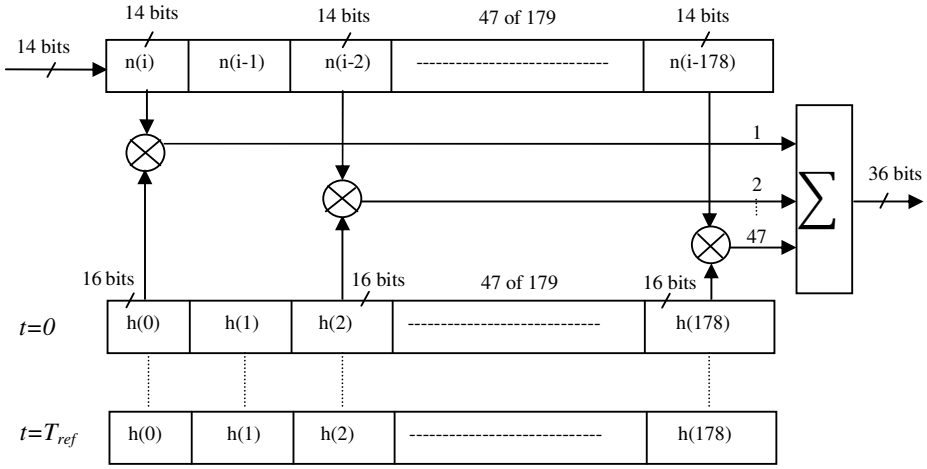

Figure 10. FIR 179 filter with 47 multipliers for a SISO channel

The general formula for FIR 179 filter with 47 multipliers is:

$$y_q(i) = \sum_{k=1}^{47} h_q(i_k).x_q(i - i_k)\ ,\ i\ \epsilon\ N \tag{26}$$

In this relation, the index q suggests the use of quantified samples and $h_q(i_k)$ is the attenuation of the k^{th} path with the delay i_kT_s.

3.3. Implementation of Each Architecture

3.3.1 Description

Figure 11 shows the XtremeDSP Virtex-IV board from Xilinx [3] used for the implementation of each architecture. This prototyping board is described in [23].

The simulations and synthesis are made with Xilinx ISE [3] and ModelSim software [33].

The XtremeDSP features dual-channel high performance ADCs (AD6645) and DACs (AD9772A) with 14-bit resolution, a user programmable Virtex-IV FPGA, programmable clocks, support for external clock, host interfacing PCI, two banks of ZBT-SRAM, and JTAG interfaces.

Figure 11. XtremeDSP Development board Kit-IV for Virtex-IV

This development kit is built with a module containing the Virtex-IV SX35 component, selected to correspond to the complexity constraints. It contains a number of arithmetic blocks (DSP blocks) which makes it possible to implement many functions occupying most of the component. This device enables us to implement different time domain or frequency domain architectures and thus to reprogram the component according to the selected (indoor or outdoor) environment.

Frequency domain or time domain MIMO 2×2 channel architectures are implemented in the FPGA Virtex-IV. To test a higher order MIMO channel, the use of more performing FPGA as Virtex-VII [3] is mandatory.

3.3.2 Implementation

A) Implementation of the Frequency Domain Architecture

Four SISO channels are needed to be implemented for a one-way 2×2 MIMO radio channel. The V4-SX35 development board utilization summary shows that in frequency domain, four SISO channels using 512 FFT/IFFT modules occupy more than 15,360 slides on the FPGA. Thus, more than 100 % of the slices are needed. Hence, it is impossible to simulate 4 SISO channels in frequency domain.

B) Implementation of Time Domain Architecture

Table 2 shows the device utilization in one V4-SX35 for four SISO channels using four FIR filters: FIR 179 with 47 multipliers, FIR 174 with 41 multipliers, FIR 161 with 43 multipliers and FIR 181 with 48 multipliers.

The time domain architecture is better in term of occupation on the FPGA. Moreover, in [23] and [24] we have shown that the time domain architecture has two other advantages: a higher SNR and a much lower latency.

Thus, in this work, the time domain architecture is used for the tests. Also, solutions are proposed to modify the number of bits used in this architecture to decrease the latency and the occupation on FPGA.

Table 2. FPGA Occupation for 4 FIR Filters

Device Utilization Summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	5,614	30,720	19%
Number of 4 input LUTs	11,688	30,720	39%
Number of occupied slices	**10,321**	**15,360**	**68%**
Number of slices containing only related logic	10,321	10,321	100%
Number of slices containing unrelated logic	0	10,321	0%
Total number of 4 input LUTs	**11,688**	**30,720**	**39%**
Number used as logic	1150		
Number used as a route-thru	1		
Number used as shift registers	179		
Number of bonded IOBs	**40**	**448**	**8%**
Number of BUFG/BUFGCTRLs	**1**	**32**	**3%**
Number used as BUFGs	**1**		
Number of FIFO16/RAMB16s	**179**	**192**	**94%**
Number used as RAMB16s	179		
Number of DSP48s	**179**	**192**	**94%**
Average Fan-out of Non-Clock Nets	1.96		

4. Implementation of the Impulse Responses in the Digital Block

4.1. Description

The channel impulse responses are stored on the hard disk of the computer and read via the PCI bus and then stored in the FPGA dual-port RAM. Figure 12 shows the connection between the computer and the FPGA board to reload the coefficients.

500 successive profiles are considered for the test of a 2×2 MIMO time-varying channel.

We simulate a shipboard indoor environment where v (the speed of moving people) is between 0 and 4 km/h. Therefore, for the test, we choose to analyze the two extreme cases. To obtain non-null Doppler frequency, two speeds are considered: 0.5 km/h and 4 km/h. The Doppler spread f_d is equal to the carrier frequency f_c multiplied by v and divided by the celerity c. The refresh frequency f_{ref} is chosen to be more than twice f_d to respect Nyquist-

Shannon theorem. With f_c = 2.2 GHz, f_{ref} is chosen to be 2.5 Hz (for v = 0.5 km/h) and 20 Hz (for v = 4 km/h).

For v = 0.5 km/h, the refreshing period is 0.4 s during which we must refresh all of the four profiles, i.e. 47 + 41 + 43 + 48 = 179 words of 16 bits = 358 bytes to transmit for a profile, which is: 358×2.5= 0.895 kBps. For v = 4 km/h, the refreshing period is 50 ms during which we must refresh all of the four SISO profiles. The amount of data transmitted for a MIMO profile is: 358/0.4= 7.16 kBps.

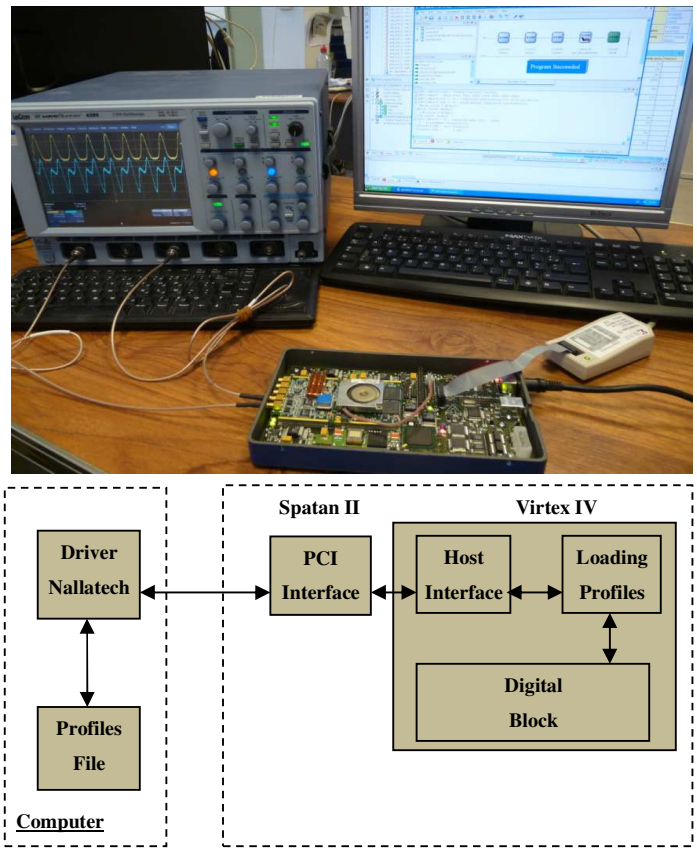

Figure 12. Connection between the computer and the XtremeDSP board

The profiles of 16 bits are stored in a text file on the hard disk of a computer. This file is then read to load the memory block which will supply the RAM blocks on the simulator (one block for each tap of the impulse response). Reading the file can be done either from USB or PCI interfaces, both available on the used prototyping board. The PCI bus is chosen to load the profiles of frequency responses. It has a speed of 30 MBps. In addition, the PCI bus is a bus of 32 bits. Thus, on each clock pulse, two samples of the impulse response are transmitted.

The Nallatech driver presented in Figure 12 provides an IP called "Host Interface" that reads the data from the PCI bus and stores these data into the FIFO memory of the IP. The module called "Loading profiles" reads and distributes the impulse responses in RAM blocks. This module called "BOX RAM" is the block "Memory" of the time domain architecture.

While a MIMO profile is used, the following MIMO profile is loaded and will be used after the refresh period.

4.2. Accuracy of the Architecture

In order to determine the accuracy of the digital block, a comparison is made between the theoretical and the Xilinx output signal. In order to test the time domain architecture, a specific input Gaussian signal $x(t)$ is considered. This input signal presented in Figure 13 is long enough to be used in streaming mode (the use of a Gaussian signal is preferred because it has a limited duration in both time and frequency domains):

$$x(t) = \begin{cases} x_{m1} e^{-\frac{(t-m_{x1})^2}{2\sigma_1^2}} & 0 \le t \le 3W_t/4 \\ -x_{m2} e^{-\frac{(t-m_{x2})^2}{2\sigma_2^2}} & 3W_t/4 \le t \le 3W_t/2 \end{cases} \quad (27)$$

where $n = 200$ (chosen greater than the length of the FIR filters to test it in streaming mode), $W_t = n.T_s$, $m_{x1} = 3.W_t/8$, $m_{x2} = 6.W_t/5$ and $\sigma_1 = \sigma_2 = m_{x1}/12$ (small enough to show the effect of each path of the impulse response on the output signal).

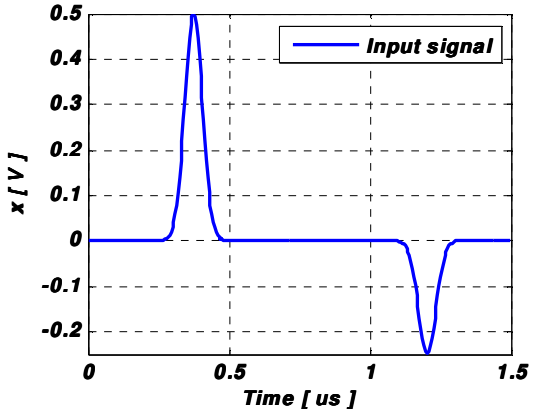

Figure 13. Input signal

The A/D and D/A converters of the development board have a full scale of $[-V_m, V_m]$, with $V_m = 1$ V. For the simulations we consider $x_{m1} = V_m/2$ and $x_{m2} = V_m/4$.

The theoretic output signals are calculated by:

$$y_1(t) = \sum_{k=1}^{47} h_{11}(i_k).x(t - i_k t_s) + \sum_{k=1}^{43} h_{21}(j_k).x(t - j_k t_s) \quad (28)$$

$$y_2(t) = \sum_{k=1}^{41} h_{12}(p_k).x(t - p_k t_s) + \sum_{k=1}^{48} h_{22}(l_k).x(t - l_k t_s) \quad (29)$$

The relative error is given for each output sample by:

$$\varepsilon(i) = \frac{Y_{xilinx}(i) - Y_{theory}(i)}{Y_{theory}(i)}.100\ [\%] \quad (30)$$

where Y_{Xilinx} and Y_{theory} are vectors containing the samples of corresponding signals. The Signal-to-Noise Ratio (SNR) is:

$$\mathrm{SNR}(i) = 20.log_{10}\left|\frac{Y_{theory}(i)}{Y_{xilinx}(i) - Y_{theory}(i)}\right|\ [dB] \quad (31)$$

where $i = \overline{1,\ 3N + \iota_{Final}}$ and i_{Final} is the largest number between the last tap of h_{11} and the last tap *of* h_{21} or between the last tap *of* h_{12} and the last tap *of* h_{22}.

After the D/A converter, the signal is limited to $[-V_m, V_m]$ with $V_m = 1$. If $|y_{max}| > 1$ V as shown in Figure 14, a reconfigurable analog amplifier placed after the DAC must multiply the signal with 2^{k_0}, where k_0 is the smallest integer verifying $|y_{max}| < 2^{k_0}$.The relative error is high only for small values of the output signal.

The global values of the relative error and of the SNR computed for the output signal before and after the final truncations are necessary to evaluate the accuracy of the architecture. The global relative error is computed by:

$$\varepsilon = \frac{\|E\|}{\|Y_{theory}\|} \times 100\ [\%] \quad (32)$$

The global SNR is computed by:

$$SNR_g = 20 \times log_{10}\frac{\|Y_{theory}\|}{\|E\|}\ [dB] \quad (33)$$

where $E = Y_{Xilinx} - Y_{theory}$ is the error vector.

For a given vector $X = [\ x_1\ ,\ x_2,\ \ldots,\ x_L\ ]$, its Euclidean norm $\|\ x\ \|$ is:

$$\|x\| = \sqrt{\frac{1}{L}\sum_{k=1}^{L} x_k^2} \tag{34}$$

Figure 14 presents the Xilinx output signal, the SNR and the relative error.

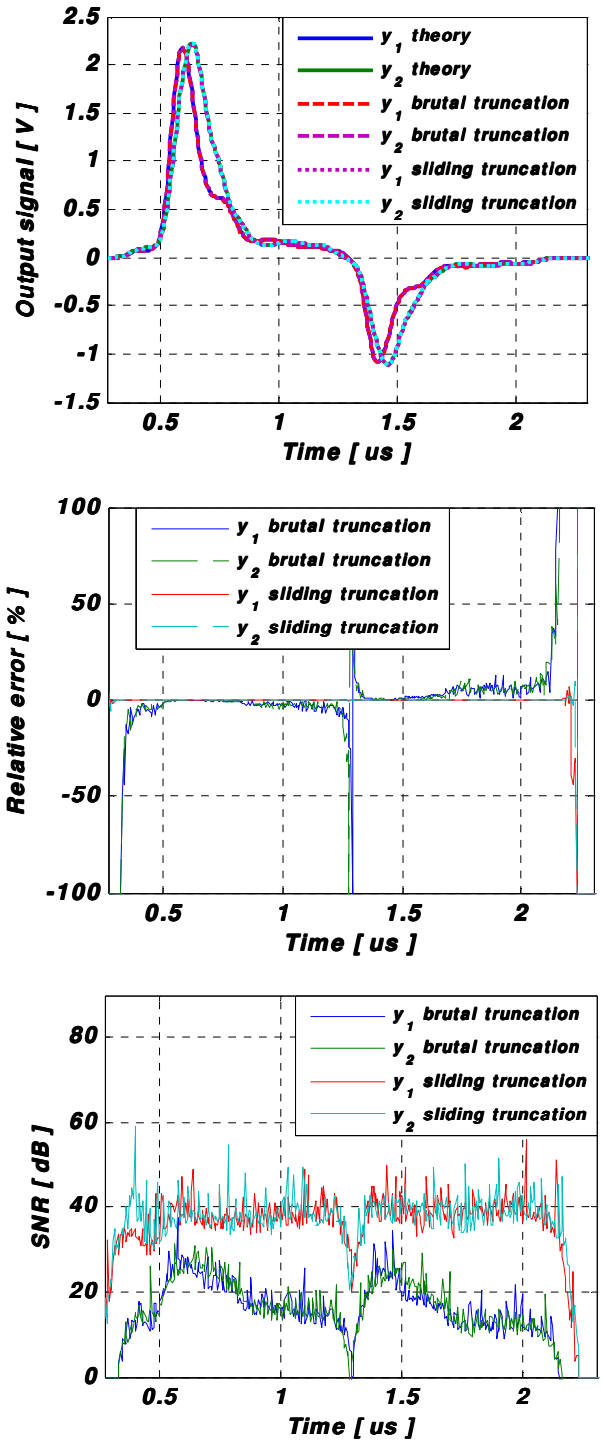

Figure 14. The Xilinx output signals, the SNR and the relative error

Table 3 shows the global values of the relative error and the SNR between the Xilinx output signal and the theoretical output signal using MIMO 2×2 time domain architecture. The results are given without truncation, with sliding window and with brutal truncations.

Table 3. The global relative error and the global SNR

Output	Error (%)	SNR (dB)
without truncation		
$\mathbf{y_1}$	0.0107	79.43
$\mathbf{y_2}$	0.0110	79.13
with sliding window truncation		
$\mathbf{y_1}$	0.0165	75.64
$\mathbf{y_2}$	0.0171	75.33
with brutal truncation		
$\mathbf{y_1}$	0.7707	42.25
$\mathbf{y_2}$	0.6738	43.42

4.3. Global Error Variation with Time-Varying Profiles

To test the simulator with time-varying 2×2 MIMO channels, 500 successive profiles are considered. For an environmental speed of 0.5 km/h, the refresh frequency f_{ref} = 2.5 Hz. Therefore, the time to simulate the 500 profiles is 200 s. Figure 15 gives the time variation of the Average Global Relative Error (AG RE) and the Average Global SNR (AG SNR) of y_1 and y_2 for the 500 successive profiles.

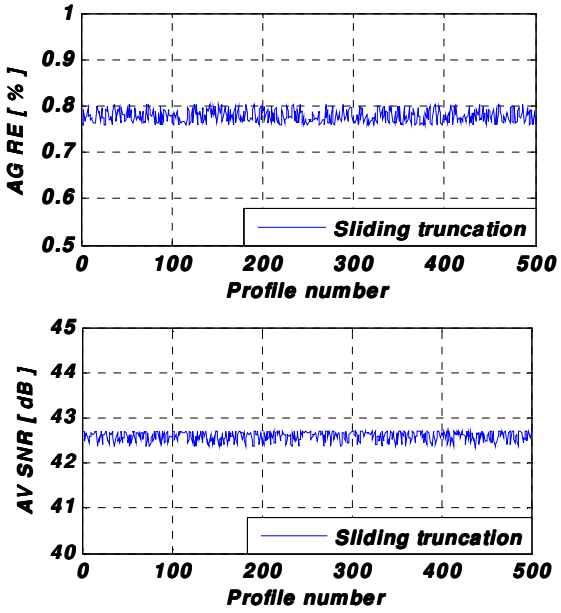

Figure 15. AG RE and AG SNR for v= 0.5 km/h

For an environmental speed of 4 km/h, f_{ref} = 20 Hz. Therefore, the time to simulate the 500 profiles is 25 s. Figure 16 shows the time variation of the AG RE and the AG SNR of y_1 and y_2 for the 500 successive profiles.

For v = 0.5 km/h, the variation of SNR is 2.03 dB. For v = 4 km/h, the variation of SNR is 2.37 dB. Therefore, after several variation of the environmental speed between 0 and 9 km/h (the maximum for an indoor environment), we conclude that the rate of variation of the SNR and the global error is related proportionally to the speed environment.

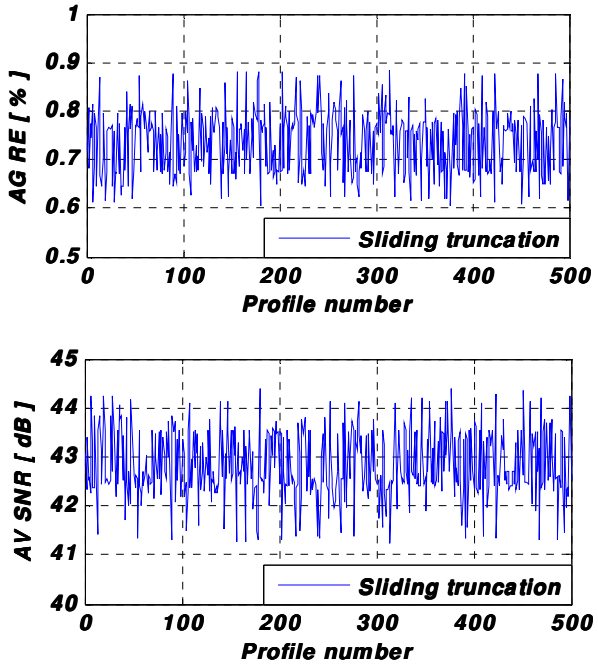

Figure 16. AG RE and AG SNR for v= 4 km/h

5. Improvement Solutions

The goal is to improve: the precision, the FPGA occupation and the latency.

Using a Normalization Factor (*NF*) at the input signal decreases significantly the error. Also, decreasing the number of bits of the implemented impulse responses presented by *h* in Figure 10 will decrease the occupation of slices in the FPGA and the latency.

5.1. Normalization Factor (*NF*)

After analyzing the relative error in Figure 14, we conclude that it is high only for small values of the output signal. Therefore, to decrease the error, a solution is proposed which consists on considering two thresholds: *SH* and *SL*. *SH* is equal to x_{max}. In fact, the input and the output signals are limited to $[-V_m,V_m]$ with V_m = 1 V. Thus, x_{max} = 0.5 V to leave a

sufficient margin for the input signal. *SL* is considered equal to 0.125 V (higher than 0.125 V the SNR is high as presented in Fig. 13 and Fig. 14). If $|x(t)| > SH$, the signal is divided by $NF = 2$. If $|x(t)| < SL$, the signal is multiplied by $NF = 2^n$ where n is the biggest integer verifying $SH < 2^n$. Two digital signals are considered: *NF* at the input of the FPGA and NF_{out} at the output. A FPGA Virtex-IV provides 28 bits as digital I/O. Thus, *NF* and NF_{out} are presented on 14 bits. Using a brutal truncation, $NF_{out} = NF$. However, using a sliding truncation, $NF_{out} = NF - STF$ where *STF* is the sliding truncation factor. NF_{out} is related to a reconfigurable analog amplifier placed after the DAC to obtain the correct output signal.

Figure 17 presents the output signal before the DAC converter.

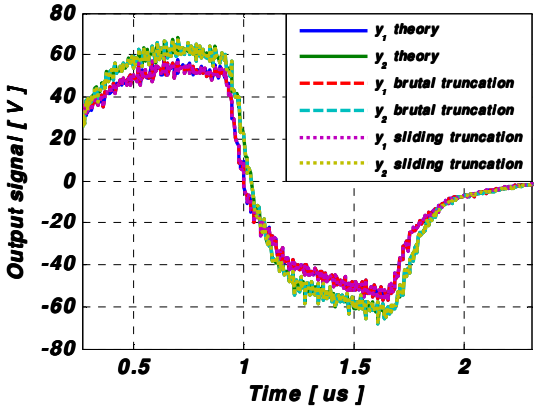

Figure 17. Output signal using *NF*

Figure 18 presents the relative error and the SNR between the Xilinx output signal and the theoretical output signal.

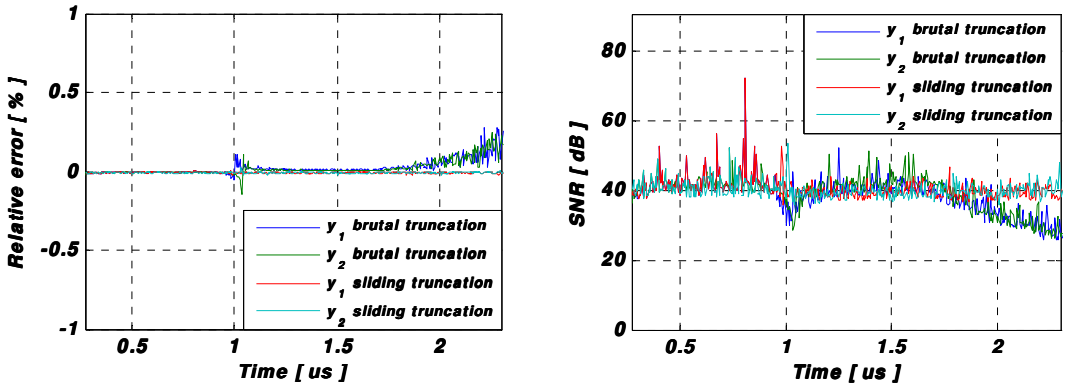

Figure 18. Relative error and SNR after adding *NF*

We notice that after adding the *NF*, the relative error decrease significantly. Table 4 presents the new values of the global relative error and the global SNR.

Table 4. The global relative error and the global SNR

Output	Error (%)	SNR (dB)
without truncation		
y_1	0.00054037	105.34
y_2	0.00046892	106.57
with sliding window truncation		
y_1	0.0104	79.62
y_2	0.0118	78.53
with brutal truncation		
y_1	0.0119	78.46
y_2	0.0100	80.02

5.2. The Error Versus the Number of Bits of *h*

To decrease the occupation of slices on the FPGA of the time domain architecture, we decrease the number of bits of *h*. A study of the average global relative error in function of the number of bits of *h* is given in Figure 19.

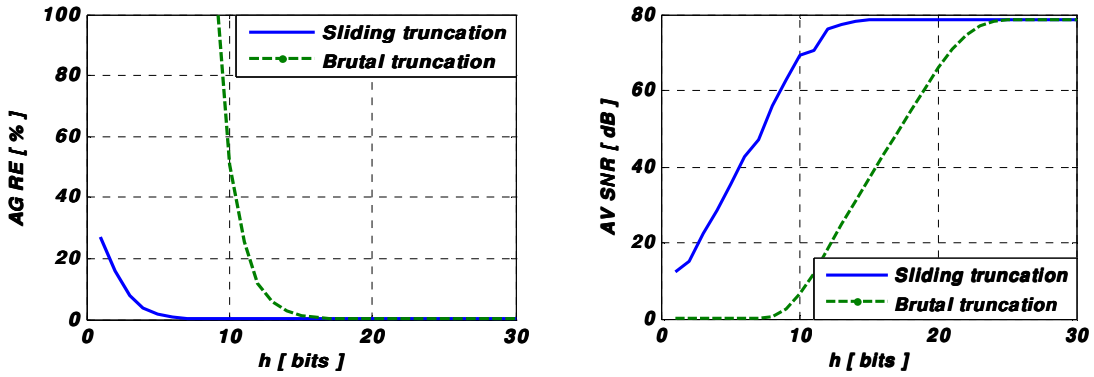

Figure 19. AG RE and AV SNR versus the number of bits of h

We can conclude that for a number of bits for *h* bigger than 5 bits, the AG RE is acceptable and the AV SNR is more than 40 dB. For a number of bits for *h* equal 6 bits, the occupation on the FPGA is reduced from 19 % to 17 %. However, the average global error using a brutal truncation exceeds 100 %, while with a sliding truncation it is 0.77 % which is acceptable. Thus, the sliding truncation is mandatory to use in this case. The *STF* is quantified on 28 bits and feeds a reconfigurable analog amplifier placed after the DAC to obtain the correct output signal. The amount of data transmitted for a profile is also reduced. In fact, the PCI bus is a bus of 32 bits. So on each clock pulse five simples of the impulse response are transmitted (instead of two simples).

Figure 20 presents the output signal, the relative error and the SNR using 6 bits for h with sliding window truncation.

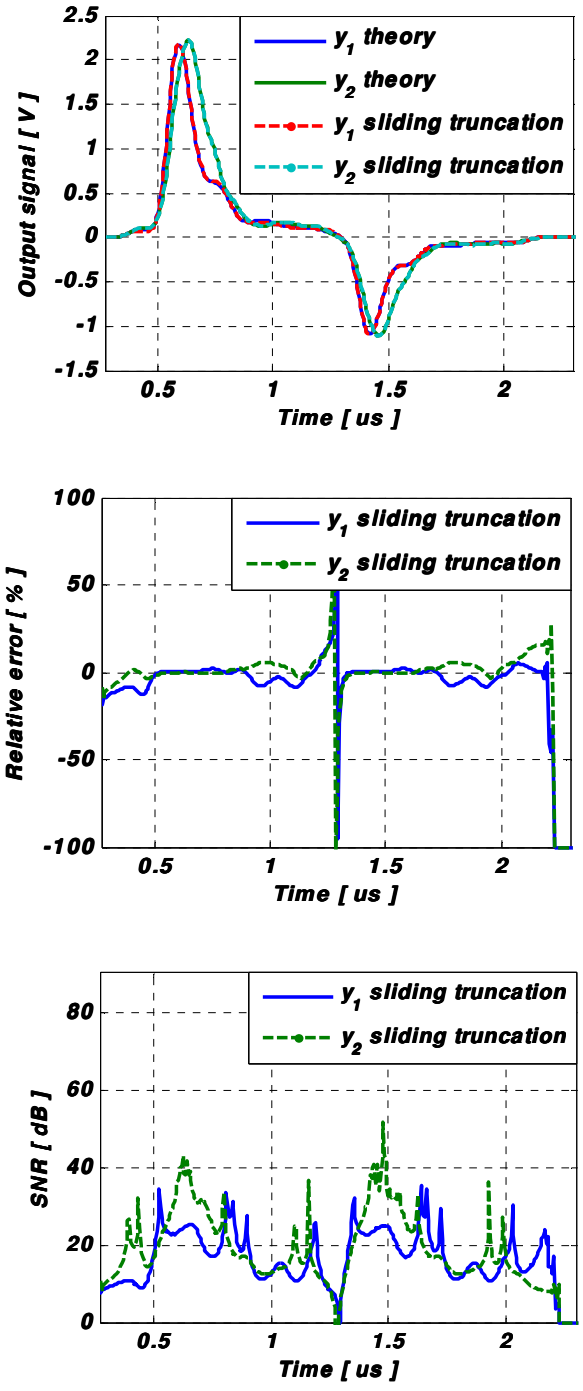

Figure 20. Output signal, relative error and SNR for 6 bits for h

The number of bits at the output before the truncation is related to the number of bits of h:

$$n_y = n_h + n_x + n_t \tag{35}$$

where n_y is the number of bits at the output, n_h is the number of bits of h, n_x = 14 is the number of bits of the input signal and n_t can be expressed by:

$$n_t = \lceil log_2(n_{tap}) \rceil \tag{36}$$

where n_{tap} is the number of taps.

Table 5 summarizes the global relative error and the global SNR using 6 bits for h.

Table 5. The global relative error and the global SNR

Output	Error (%)	SNR (dB)
without truncation		
$\mathbf{y_1}$	0.9982	40.05
$\mathbf{y_2}$	0.5482	45.21
with sliding window truncation		
$\mathbf{y_1}$	0.9994	40.04
$\mathbf{y_2}$	0.5483	45.21
with brutal truncation		
$\mathbf{y_1}$	579.3240	0.2053
$\mathbf{y_2}$	541.2234	0.2282

By reducing the number of bits of h from 14 to 6, we reduce the occupation on the FPGA.

6. Conclusion

In this paper, the measured impulse responses of a 2×2 MIMO propagation channel obtained by measurement on a shipboard environment have been presented. These impulse responses have been used in the digital block of a hardware simulator.

After a comparative study, the time domain architecture used for the design of the digital block represents the best solution, especially for MIMO systems. In fact, it occupies just 19 % of slices on the FPGA Virtex-IV. Also, it generates a small latency of 115 ns.

Moreover, a study of the precision of the architecture for time-varying 2×2 MIMO channels has been presented. It has been shown that the global relative error does not exceed 0.9 %. Therefore, time-varying impulse responses can be used by the architecture.

Lastly, in order to reduce the error of the output signals and the occupation on the FPGA, two improvement solutions have been presented. The first uses a normalizing factor. It reduces the global output relative error from 0.7707 % to 0.0119 % using brutal truncation. The second varies the number of bits of impulse response. It reduces the occupation of slices on the FPGA from 19 % to 17 %.

Simulations made using a Virtex-VII [3] XC7V2000T platform will allow us to simulate up to 300 SISO channels. Measurement campaigns will also be carried out with the MIMO channel sounder realized by IETR, for various types of environments. A graphical user interface will also be designed to allow the user to select the channel model and to select the channel parameters. The final objective of these measurements is to obtain realistic and reliable impulse responses of the MIMO channel in order to supply the digital block of the hardware simulator.

Acknowledgments

The authors would like to thank the “Région Bretagne” for its financial support of this work which is a part of PALMYRE-II project.

References

[1] S. Behbahani, R. Merched, A. Eltawil, "Optimizations of a MIMO relay network", IEEE Trans. on Signal Processing, vol. 56, no. 10, pp. 5062–5073, Oct. 2008.

[2] A. Cetiner, E. Sengul, E. Akay, E. Ayanoglu, "A MIMO system with multifunctional reconfigurable antennas", IEEE Antennas Wireless Propag. Lett., vol. 5, no. 1, pp. 463–466, Dec. 2006.

[3] "Xilinx: FPGA, CPLD and EPP solutions", www.xilinx.com.

[4] S. Picol, G. Zaharia, D. Houzet, G. El Zein, "Design of the digital block of a hardware simulator for MIMO radio channels", IEEE PIMRC, Helsinki, Finland, Sep. 2006.

[5] V. Erceg, L. Shumacher, P. Kyritsi, et al., "TGn Channel Models", IEEE 802.11- 03/940r4, May 10, 2004.

[6] Agilent Technologies, "Advanced design system – LTE channel model - R4-070872 3GPP TR 36.803 v0.3.0", 2008.

[7] H. Farhat, R. Cosquer, G. Grunfelder, L. Le Coq, G. El Zein, "A dual band MIMO channel sounder at 2.2 and 3.5 GHz", Instrumentation and Measurement Technology Conference Proceedings, Victoria, BC, Canada, May 2008.

[8] P. Almers, E. Bonek et al., "Survery of Channel and Radio Propagation Models for Wireless MIMO Systems", EURASIP Journal on Wireless Communications and Networking, Article ID 19070, 2007.

[9] J. Salz, J. H. Winters, "Effect of fading correlation on adaptive arrays in digital mobile radio", IEEE Trans. on Vehicular Technology, vol. 43, no. 4, pp. 1049-1057, Nov. 1994.

[10] L. Schumacher, K. I. Pedersen, P.E. Mogensen, "From antenna spacings to theoretical capacities – guidelines for simulating MIMO systems", in Proc. PIMRC Conf., vol. 2, pp. 587-592, Sept. 2002.

[11] "Wireless Channel Emulator", Spirent Communications, 2006.

[12] "Baseband Fading Simulator ABFS, Reduced costs through baseband simulation", Rohde & Schwarz, 1999.

[13] P. Murphy, F. Lou, A. Sabharwal, P. Frantz, "An FPGA Based Rapid Prototyping Platform for MIMO Systems", Asilomar Conf. on Signals, Systems and Computers, ACSSC, vol. 1, pp. 900-904, 9-12 Nov. 2003.

[14] P. Murphy, F. Lou, J. P. Frantz, "A hardware testbed for the implementation and evaluation of MIMO algorithms", Conf. on Mobile and Wireless Communications Networks, Singapore, Oct. 2003.

[15] S. Buscemi, R. Sass, "Design of a scalable digital Wireless Channel Emulator for networking radios", Military Communications Conference, MILCOM, Charleston, SC, USA, Nov. 2011.

[16] D. N. Vizireanu, S. V. Halunga, "Single sine wave parameters estimation method based on four equally spaced samples", International Journal of Electronics, vol. 98, no. 7, pp. 941-948, July 2011.

[17] D. N. Vizireanu, S. V. Halunga, "Simple, Fast and Accurate Eight Points Amplitude Estimation Method of Sinusoidal Signals for DSP based Instrumentation", Journal of Instrumentation, JINST 7, vol. 7, pp. 1-10, P04001, April 2012.

[18] S. Picol, G. Zaharia, D. Houzet, G. El Zein, "Hardware simulator for MIMO radio channels: Design and features of the digital block", Proc. of IEEE VTC-Fall, Calgary, Canada, Sept. 2008.

[19] Umansky, M. Patzold, "Design of Measurement-Based Stochastic Wideband MIMO Channel Simulators", IEEE Globecom, Honolulu, Nov. 2009.

[20] H. Eslami, S. V. Tran, A. M. Eltawil, "Design and implementation of a scalable channel Emulator for wideband MIMO systems", IEEE Trans. on Vehicular Technology, vol. 58, no. 9, pp. 4698-4708, Nov. 2009.

[21] S. Fouladi Fard, A. Alimohammad, B. Cockburn, C. Schlegel, "A single FPGA filter-based multipath fading emulator", Globecom, Honolulu, Nov. 2009.

[22] M. Al Mahdi Eshtawie, M. Bin Othma, "An algorithm proposed for FIR Filter coefficients representation", World Academy of Science, Engineering and Technology, 2007.

[23] B. Habib, G. Zaharia, G. El Zein, "Improved Frequency Domain Architecture for the Digital Block of a Hardware Simulator for MIMO Radio Channels", IEEE ISSCS, Iasi, Romania, June 2011.

[24] B. Habib, G. Zaharia, G. El Zein, "MIMO Hardware Simulator: Digital Block Design for 802.11ac Applications with TGn Channel Model Test", IEEE VTC Spring, Yokohama, Japan, May 2012.

[25] B. Habib, G. Zaharia, G. El Zein, "Digital Block Design of MIMO Hardware Simulator for LTE Applications", IEEE ICC, Ottawa, Canada, June 2012.

[26] K. Borries, E. Anderson, P. Steenkiste, "Network-Scale Emulation of General Wireless channels", IEEE VTC Fall, San Francisco, USA, Sep. 2011.

[27] R. Roy, T. Kailath, "ESPRIT – estimation of signal parameters via rotational invariance techniques", IEEE Trans. on Acoustics, Speech and Signal Processing, vol. 37, no. 7, pp. 984-995, July 1987.

[28] J. A. Fressler, A. O. Hero, "Space-alternating generalized expectation-maximization algorithm", IEEE Trans. on Signal Processing, vol. 42, no. 10, pp. 2664-2677, Oct. 1994.

[29] J. P. Kermoal, L. Schumacher, K. I. Pedersen, P. E. Mogensen, F. Frederiksen, "A stochastic MIMO radio channel model with experimental validation", IEEE Journal on Selected Areas of Commun., vol. 20, no. 6, pp. 1211-1226, Aug. 2002.

[30] W. C. Jakes, "Microwave mobile communications", Wiley & Sons, New York, Feb. 1975.

[31] Q. H. Spencer, et al., "Modeling the statistical time and angle of arrival characteristics of an indoor environment", IEEE J. Select. Areas Commun., vol. 18, no. 3, pp. 347-360, March 2000.

[32] C-C. Chong, D. I. Laurenson, S. McLaughlin, "Statistical Characterization of the 5.2 GHz wideband directional indoor propagation channels with clustering and correlation properties", IEEE VTC Fall, Vancouver, Canada, Sept. 2002.

[33] ModelSim - Advanced Simulation and Debugging, http://model.com.